

Chapter 4

World Model

A world model enables an agent to predict and reason about future states without direct trial-and-error in reality. This section explores how human cognitive studies on “mental models” relate to AI world models in artificial intelligence, categorizing them under four paradigms: *implicit paradigm*, *explicit paradigm*, *simulator-based paradigm*, and a class of other emergent methods (e.g., *instruction-driven paradigm*). We then discuss how world models inherently intersect with other agentic components and conclude with open questions and future directions that unite these perspectives under a unified theoretical and practical framework.

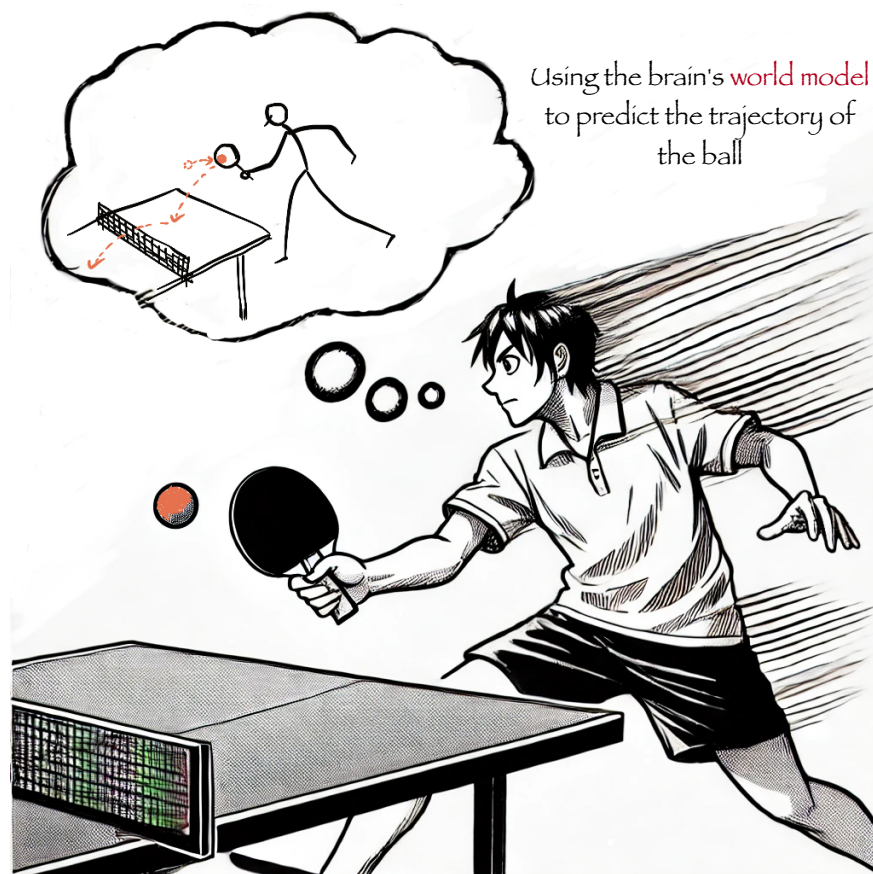


Figure 4.1: Humans can use their brain’s model of the world to predict the consequences of their actions. For example, when playing table tennis, a player can imagine or predict the trajectory of the ball after an action.

4.1 The Human World Model

Humans naturally construct internal representations of the world, often referred to as *mental models* in psychology [341, 342, 343]. These models serve as compact and manipulable depictions of external reality, enabling individuals to predict outcomes, plan actions, and interpret novel scenarios with minimal reliance on direct trial-and-error. Early work on spatial navigation, for instance, showed that humans and animals form “cognitive maps” of their surroundings [341], suggesting an underlying ability to imagine potential paths before actually traversing them.

Craik’s seminal argument was that the human mind runs internal “small-scale models of reality” [342] to simulate how events might unfold and evaluate possible courses of action. Later studies proposed that such simulations stretch across modalities—vision, language, and motor control—and are dynamically updated by comparing predictions to new observations. This process merges *memory recall* with *forward projection*, implying a close interplay between stored knowledge and the active generation of hypothetical future states [343]. More recent predictive processing theories such as “Surfing Uncertainty” [344] propose that the brain operates as a hierarchical prediction machine, continuously generating top-down predictions about sensory inputs and updating its models based on prediction errors.

Critically, these human mental models are:

- **Predictive:** They forecast changes in the environment, informing decisions about where to move or how to respond.
- **Integrative:** They combine sensory input, past experience, and abstract reasoning into a unified perspective on “what might happen next”.
- **Adaptive:** They are revised when reality diverges from expectation, reducing the gap between imagined and actual outcomes over time.
- **Multi-scale:** They operate seamlessly across different temporal and spatial scales, simultaneously processing immediate physical dynamics (milliseconds), medium-term action sequences (seconds to minutes), and long-term plans (hours to years). This flexibility allows humans to zoom in on fine-grained details or zoom out to consider broader contexts as needed.

Consider hunger and eating as an illustration of integrated world modeling. When hungry, a person’s internal model activates predictions about food—simulating not just visual appearance but tastes, smells, and anticipated satisfaction—triggering physiological responses like salivation before food is even present. This demonstrates seamless integration across perception, memory, and action planning.

The example also highlights adaptivity: once satiated, the same model dynamically updates, reducing predicted reward values for further eating. Despite recognizing the same food items, their anticipated utility changes based on internal state. Furthermore, humans maintain counterfactual simulations—declining dessert now while accurately predicting they would enjoy it later—enabling complex planning across hypothetical scenarios and time horizons, a capability comprehensive AI world models strive to replicate.

In sum, the *human world model* is not a static library of facts, but a flexible and ever-evolving mental construct, deeply rooted in perception and memory, that continuously shapes (and is shaped by) the individual’s interactions with the outside world.

4.2 Translating Human World Models to AI

Research in artificial intelligence has long sought to replicate the *predictive, integrative, and adaptive* qualities exhibited by human mental models [341, 342]. Early reinforcement learning frameworks, for instance, proposed learning an *environment model* for planning—exemplified by Dyna [345]—while contemporaneous work investigated using neural networks to anticipate future observations in streaming data [346, 347]. Both directions were motivated by the idea that an internal simulator of the world could enable more efficient decision-making than purely reactive, trial-and-error learning.

Subsequent advancements in deep learning brought the notion of “AI world models” into sharper focus. One influential approach introduced an end-to-end *latent generative model* of an environment (e.g., “World Models” [348]), whereby a recurrent neural network (RNN) and variational auto-encoder (VAE) together learn to “dream” future trajectories. These latent rollouts allow an agent to train or refine policies offline, effectively mirroring how humans mentally rehearse actions before executing them. Alongside such implicit designs, explicit forward-modeling methods emerged in model-based RL, letting agents predict $P(s' \mid s, a)$ and plan with approximate lookahead [349, 350].

Another branch of work leveraged large-scale simulators or real-world robotics to ground learning in richly diverse experiences [351, 352]. Such setups are reminiscent of how human children learn by actively exploring their environments, gradually honing their internal representations. Yet a key question lingers: can agentic systems unify these approaches (implicit generative modeling, explicit factorization, and simulator-driven exploration) into a cohesive “mental model” akin to that observed in humans? The recent proliferation of language-model-based reasoning [107, 74] hints at the potential to cross modalities and tasks, echoing how humans integrate linguistic, visual, and motor knowledge under one predictive framework.

Overall, as AI systems strive for flexible, sample-efficient learning, the *AI world model* stands as a conceptual bridge from cognitive theories of mental models to implementations that equip artificial agents with *imagination*, *predictive reasoning*, and *robust adaptation* in complex domains.

4.3 Paradigms of AI World Models

Designing an *AI world model* involves determining how an AI agent acquires, represents, and updates its understanding of the environment’s dynamics. While implementations vary, most approaches fall into four broad paradigms: *implicit*, *explicit*, *simulator-based*, and *hybrid or instruction-driven* models. These paradigms can be further analyzed along two key dimensions: reliance on *internal* (neural-based) vs. *external* (rule-based or structured) mechanisms, and overall *system complexity*. Figure 4.2 illustrates this two-dimensional space, showing how different approaches distribute themselves across these axes. Generally, implicit models tend to rely more on internal mechanisms, while explicit and simulator-based models incorporate more external structures. Simulator-based and explicit models also tend to be more complex than implicit and hybrid approaches, reflecting their structured reasoning and engineered constraints.

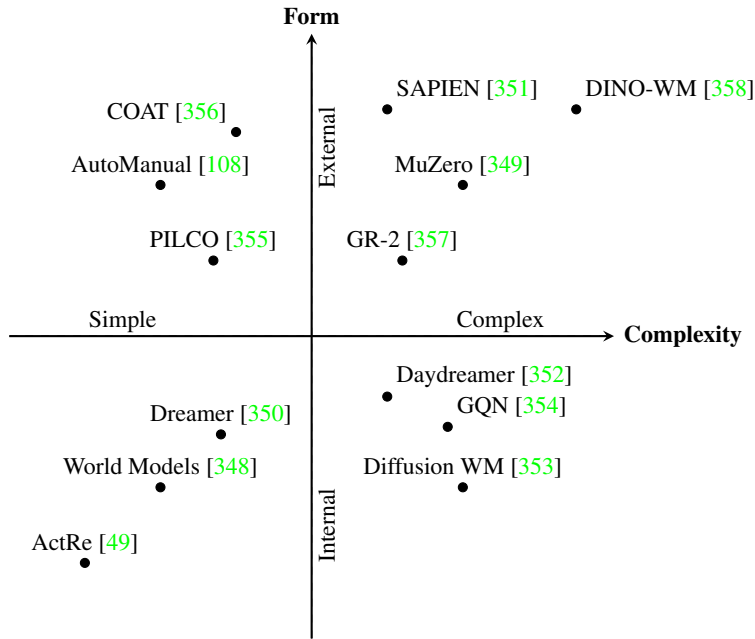


Figure 4.2: A two-dimensional layout of AI world-model methods. The horizontal axis indicates *Complexity* (left to right). The vertical axis spans *Internal* approaches (bottom) to *External* solutions (top). Approximate positions reflect each method’s reliance on large learned networks vs. explicit rules or code, and its overall system complexity.

4.3.1 Overview of World Model Paradigms

An *AI world model* is broadly any mechanism by which an agent captures or accesses approximate environment dynamics. Let \mathcal{S} denote the set of possible environment *states*, \mathcal{A} the set of *actions*, and \mathcal{O} the set of *observations*. In an idealized Markovian framework, the environment is characterized by transition and observation distributions:

$$T(s'|s, a) : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S}), \quad (4.1)$$

$$O(o|s') : \mathcal{S} \rightarrow \Delta(\mathcal{O}), \quad (4.2)$$

where $T(\cdot)$ dictates how states evolve under actions, and $O(\cdot)$ defines how states produce observations. A **world model** typically *learns* or *utilizes* approximations of these functions (or a variant), allowing the agent to *predict* future states or observations without executing real actions in the environment.

Numerous approaches exist to implement these approximations, which we group into four main **paradigms**:

- **Implicit paradigm:** A single neural network or latent structure encodes both transition and observation mappings without explicit factorization. World Models [348] or large language models used for environment reasoning are typical examples. Agents generally unroll this black-box function to simulate hypothetical trajectories.
- **Explicit paradigm:** The agent directly models or has access to learnable transition model T_θ and observation model O_θ , often enabling interpretability or modular design. Model-based RL methods—like MuZero [349] or Dreamer [350]—learn or refine T_θ , planning in an approximated state space. Generative visual models such as [353, 358] fall under this category if they explicitly predict the next states or frames.
- **Simulator-Based paradigm:** Rather than approximating (4.1)–(4.2), the agent relies on an external simulator or even the physical world as the ground-truth. Systems like SAPIEN [351] or real-robot pipelines [352] can be seen as “native” environment models that the agent queries. Although no learned $T(\cdot)$ is required, the agent pays a cost in terms of runtime or real-world risks.
- **Other paradigms (Hybrid or Instruction-Driven):** Methods that defy simple classification. They may store emergent rules in textual form [108], refine implicit LLM knowledge into partial causal graphs [356], or combine external components with learned sub-modules. Such approaches highlight the evolving nature of world-model research, where instructions, symbolic rules, or on-the-fly structures can complement more traditional approximations.

Throughout the remainder of this subsection, we examine how each paradigm addresses (or circumvents) Equations (4.1) and (4.2), the trade-offs in interpretability and scalability, and their relative merits for different tasks ranging from text-based to high-dimensional embodied control.

4.3.2 Implicit Paradigm

In the **implicit** paradigm, an agent encodes all environment dynamics—including how states evolve and how observations are generated—within a single (or tightly coupled) neural model. Formally, one maintains a latent state h_t that is updated according to

$$h_{t+1} = f_\theta(h_t, a_t), \quad \hat{o}_{t+1} = g_\theta(h_{t+1}), \quad (4.3)$$

where f_θ subsumes the transition function $T(\cdot)$ (and part of $O(\cdot)$) from Eqs. (4.1)–(4.2), but without making these components explicit. A classic example is the *World Models* framework [348], in which a Variational Autoencoder (VAE) first compresses visual inputs into latent codes, and a recurrent network predicts the next latent code, effectively “dreaming” trajectories in latent space. Recent work also explores repurposing large language models (LLMs) for environment simulation in purely textual or symbolic domains [107, 74], although these models are not always grounded in strict time-series or physics-based data.

Because implicit models fuse the transition and observation mechanisms into one monolithic function, they can be elegantly trained end to end and unrolled internally for planning. However, they tend to be opaque: it is difficult to interpret how precisely the network captures domain constraints or to inject knowledge directly into any part of the transition. This can be advantageous for highly complex environments where a single large-capacity model can discover latent structure on its own, but it also risks brittleness under distribution shifts. Overall, the implicit paradigm is appealing for its simplicity and flexibility, but it can pose challenges when interpretability, explicit constraints, or fine-grained control of the dynamics are required.

4.3.3 Explicit Paradigm

The **explicit** paradigm instead factorizes the world model, often by learning or encoding a transition function $\hat{T}_\theta(s_{t+1} | s_t, a_t)$ and an observation function $\hat{O}_\theta(o_{t+1} | s_{t+1})$. This explicit separation makes it possible to query each function independently. For instance, one might draw samples from

$$\hat{s}_{t+1} \sim \hat{T}_\theta(s_t, a_t), \quad \hat{o}_{t+1} \sim \hat{O}_\theta(\hat{s}_{t+1}). \quad (4.4)$$

Model-based reinforcement-learning algorithms like MuZero [349] or Dreamer [350] exemplify this paradigm by refining a forward model for planning. Other explicit approaches prioritize fidelity in generating future frames, such as

Diffusion WM [353], which applies diffusion processes at the pixel level, or DINO-WM [358], which rolls out future states within a pretrained feature space.

By factorizing transitions and observations, explicit methods can be more interpretable and more amenable to debugging and domain-specific constraints. That said, they are still sensitive to model errors: if \hat{T}_θ deviates significantly from reality, the agent’s planning and decision-making can become ineffective. Many explicit systems still rely predominantly on internal (neural) representations, but they may integrate external planners (e.g., tree-search algorithms) to leverage the explicit transition structure. This blend of learned and symbolic components offers a natural way to incorporate human knowledge, while preserving the strengths of deep learning.

4.3.4 Simulator-Based Paradigm

In the **simulator-based** paradigm, the agent outsources environment updates to a simulator, effectively bypassing the need to learn \hat{T}_θ from data. Formally,

$$(s_{t+1}, o_{t+1}) \leftarrow \mathcal{SIM}(s_t, a_t), \quad (4.5)$$

where \mathcal{SIM} is often an external physics engine or the real world itself. Platforms like SAPIEN [351] and AI Habitat provide deterministic 3D physics simulations, allowing agents to practice or iterate strategies in a controlled environment. Alternatively, methods such as Daydreamer [352] treat real-world interaction loops like a “simulator,” continually updating on-policy data from physical robots.

This approach yields accurate transitions (assuming the simulator accurately reflects reality), which alleviates the risk of learned-model errors. However, it can be computationally or financially expensive, especially if the simulator is high fidelity or if real-world trials are time-consuming and risky. As a result, some agents combine partial learned dynamics with occasional simulator queries, aiming to balance accurate rollouts with efficient coverage of state-action space.

4.3.5 Hybrid and Instruction-Driven Paradigms

Beyond these three primary paradigms, there is a growing number of **hybrid** or **instruction-driven** approaches, which blend implicit and explicit modeling or incorporate external symbolic knowledge and large language models. Often, these systems dynamically extract rules from data, maintain evolving textual knowledge bases, or prompt LLMs to hypothesize causal relationships that can then be tested or refined.

AutoManual [108], for example, iteratively compiles interactive environment rules into human-readable manuals, informing future actions in a more transparent way. Meanwhile, COAT [356] prompts an LLM to propose possible causal factors behind observed events, then validates or refines those factors via direct interaction, bridging text-based reasoning with partial learned models. Although these solutions offer remarkable flexibility—particularly in adapting to unfamiliar domains or integrating real-time human insights—they can be inconsistent in how they structure or update internal representations. As language-model prompting and real-time rule discovery continue to evolve, these hybrid methods are poised to become increasingly common, reflecting the need to balance end-to-end learning with the transparency and adaptability offered by external instruction.

Until now, we have introduced the four typical paradigms of existing world model techniques, as illustrated in Figure 4.3.5. As we can see, each type of technique has trade-offs in different aspects.

4.3.6 Comparative Summary of Paradigms

The table summarizes the key methods in AI world modeling, categorizing them based on their reliance on *external* or *internal* mechanisms, their complexity, and their respective paradigms. The form column uses \circ for external approaches and \bullet for internal ones, with mixed methods having both symbols. This classification aligns with the previous subsections, including the detailed discussion of each paradigm, and complements the visual representation in Figure 4.2.

4.4 Relationships to Other Modules

A comprehensive AI world model does not exist in isolation but interacts with several key components of the agent’s architecture. These include (but not limited to) the memory, perception, and action modules. In this subsection, we explore how world models integrate with these critical components to enable coherent and adaptive behavior in dynamic environments.

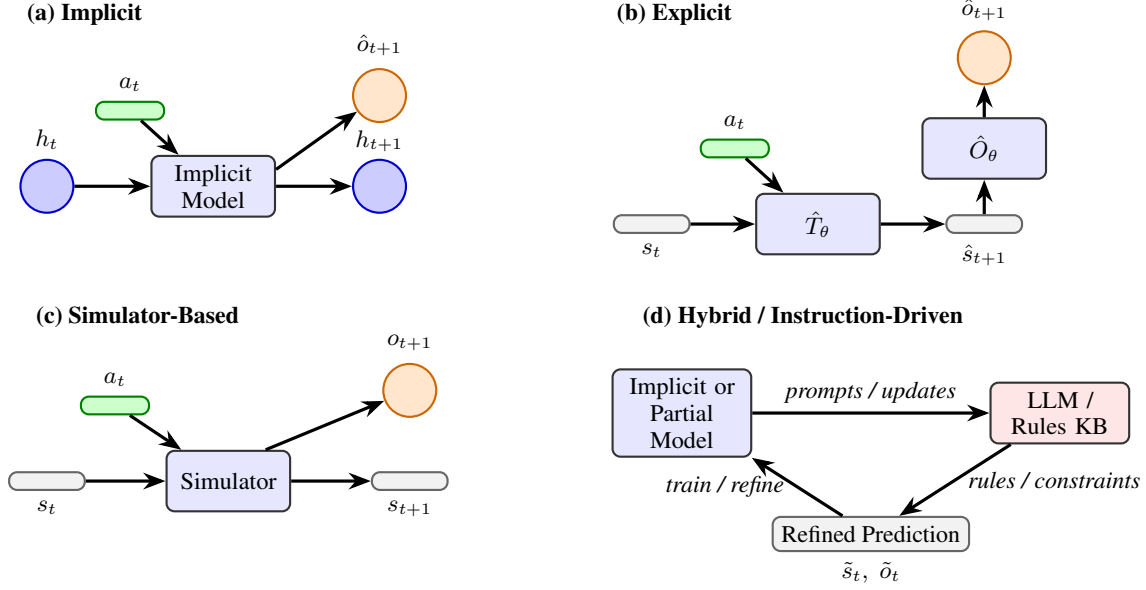


Figure 4.3: Four paradigms of world modeling: (a) implicit, (b) explicit, (c) simulator-based, and (d) hybrid/instruction-driven.

Table 4.1: Summary of AI world-model methods across paradigms, showing their form (External or Internal), complexity, and paradigm.

Method	Form	Complexity	Paradigm
ActRe [49]	•	Simple	Implicit
World Models [348]	•	Simple	Implicit
Dreamer [350]	•	Moderate	Implicit
Diffusion WM [353]	•	High	Explicit
GQN [354]	•	High	Explicit
Daydreamer [352]	○	High	Simulator-based
SAPIEN [351]	○	High	Simulator-based
PILCO [355]	○	Moderate	Explicit
AutoManual [108]	○	Simple	Other
MuZero [349]	○	High	Explicit
GR-2 [357]	•	High	Explicit
DINO-WM [358]	•	High	Explicit
COAT [356]	○	Moderate	Other

4.4.1 Memory and the World Model

Memory systems play a crucial role in the operation of world models. While a world model generates predictive representations of future states or actions, memory serves as the foundation upon which these representations are built and updated. The relationship between the world model and memory can be viewed as a loop where the world model predicts potential futures, while the memory stores past experiences, observations, and learned patterns, allowing for context-dependent reasoning and future predictions.

Memory mechanisms can be structured in various ways, including:

- **Short-term memory:** This enables the agent to hold and update its internal state temporarily, storing the most recent interactions or observations. This short-term context helps the agent make decisions in the immediate environment.
- **Long-term memory:** This serves as a more persistent repository of experiences and general knowledge about the environment. A world model can interact with long-term memory to refine its predictions, and it may use historical data to make more informed decisions or simulate more realistic futures.

For example, in model-based RL frameworks like Dreamer [350], recurrent neural networks act as both the world model and a form of memory, maintaining a latent state that is updated with each time step to predict future states. This form of integrated memory allows the agent to both recall past interactions and anticipate future ones.

4.4.2 Perception and the World Model

Perception refers to the agent’s ability to sense and interpret its environment through various modalities (e.g., vision, touch, sound, etc.). The world model relies heavily on accurate sensory input to form coherent predictions about the environment. In many AI systems, the perception module converts raw sensor data into a higher-level representation, such as an image, sound wave, or other structured data.

A key aspect of the interaction between the world model and perception is how the agent processes and integrates sensory input into the model. The world model often depends on processed data (such as features from convolutional neural networks or embeddings from transformers) to simulate potential futures. Additionally, the world model can guide perceptual processes by focusing attention on the most relevant sensory input needed to refine predictions.

For example, in autonomous robotics, perception systems typically detect objects or environmental features, which are then fed into a world model that predicts how the scene will evolve. RoboCraft [359] achieves this perception-to-modeling transformation by converting visual observations into particles and capturing the underlying system structure through graph neural networks. PointNet [360] further enriches perception systems’ understanding of physical space by encoding unstructured 3D point clouds to capture spatial characteristics of the environment. In navigation tasks, OVER-NAV [361] further combine large language models and open-vocabulary detection to construct the relationship between multi-modal signals and key information, proposing an omni-graph to capture the structure of local space as the world model for navigation tasks. This feedback loop between perception and the world model enables agents to update their perception dynamically based on ongoing predictions, allowing for real-time adaptation.

4.4.3 Action and the World Model

Action refers to the decision-making process through which an agent interacts with its environment. In agentic systems, actions are driven by the world model’s predictions of future states. The world model aids in planning by simulating the outcomes of different actions before they are executed, allowing the agent to choose the most optimal course of action based on the predicted consequences.

The integration between world models and action modules can take various forms:

- **Model-based planning:** World models explicitly model the environment’s transition dynamics [349, 362, 107], allowing the agent to simulate multiple action sequences (rollouts) before selecting the most optimal one.
- **Exploration:** World models also support exploration strategies by simulating unseen states or unexpected actions [363, 350, 364]. These simulations enable the agent to evaluate the potential benefits of exploring new parts of the state space.

In model-based planning, MuZero [349] performs implicit planning through self-play and Monte Carlo Tree Search (MCTS), transforming current state representations into future state and reward predictions to guide the decision-making process without prior knowledge of environment rules. In contrast, MPC [362] utilizes explicit dynamics models to predict multiple possible trajectories within a finite time horizon, determines the optimal control sequence by solving an optimization problem, and continuously updates planning using a receding horizon approach. Alpha-SQL [365], on the other hand, integrates an LLM-as-Action-Model within an MCTS framework to explore potential SQL queries within the database’s “world model”. This approach dynamically generates promising SQL construction actions based on partial query states, enabling zero-shot Text-to-SQL interactions without task-specific fine-tuning. Unlike MuZero, which focuses on planning for decision-making in uncertain environments, Alpha-SQL applies MCTS in a specific task—guiding SQL query construction through self-generated actions within a complex database context.

For exploration strategies, Nagabandi et al. [363] incentivizes agents to explore unknown regions by providing reward mechanisms (exploration bonuses) for discovering new states. Dreamer [350] propose that world models can generate imaginary action sequences (imaginary rollouts), allowing agents to safely evaluate the benefits of new actions in simulated environments without risking real-world experimentation. Similarly, in the discrete world model Hafner et al. [364], agents efficiently explore complex environments by simulating multiple possible future states, effectively balancing the trade-off between exploration and exploitation.

For example, in reinforcement learning, agents can employ a learned world model to simulate future trajectories in action-selection tasks. The world model evaluates the potential rewards of different actions, enabling the agent to plan effectively and take actions that maximize long-term goals.

4.4.4 Cross-Module Integration

While memory, perception, and action are discussed as separate modules, the true strength of world models lies in their ability to seamlessly integrate across these domains. A world model continuously receives sensory input, updates its internal memory, simulates future states, and uses this information to drive action selection. The iterative feedback loop between these modules allows agents to engage in intelligent, goal-directed behavior that is highly adaptive to changes in the environment.

This cross-module interaction is particularly relevant in complex, dynamic systems such as robotics, where an agent must continuously adapt its internal representation of the world, process sensory input, store relevant experiences, and take actions in real time. In the context of embodied agents, the integration of these modules ensures that predictions made by the world model are grounded in current observations and the agent’s ongoing experiences.

World models provide a fundamental unifying principle across modalities. Whether predicting physical outcomes in embodied robotics, anticipating visual changes on screens, or inferring semantic relationships in text, the core mechanism remains consistent: generating predictions about how states evolve under different actions. This cross-modal capacity explains why humans transition effortlessly between manipulating objects, navigating interfaces, and processing language—all activities driven by the same underlying predictive architecture. Future AI systems may achieve similar integration by developing world models that bridge these traditionally separate domains through a common predictive framework.

In summary, the relationship between the world model and the other modules—memory, perception, and action—forms the backbone of intelligent behavior in AI systems. Each module contributes to a cycle of prediction, update, and action, allowing agents to function effectively in dynamic and uncertain environments. These interactions highlight the need for a holistic approach when designing agent architectures, where world models are closely intertwined with sensory input, memory systems, and decision-making processes.

4.5 Summary and Discussion

The evolution of AI world models, from early cognitive insights to advanced AI architectures, underscores the growing realization that true intelligence relies on the ability to predict, simulate, and imagine. Unlike classical reinforcement learning, where agents operate solely through trial-and-error interactions, world models enable foresight—agents can plan, anticipate, and adapt to changes before they happen. This leap in cognitive modeling—whether implicit, explicit, or simulator-based—marks a significant shift in how machines can be endowed with flexibility, robustness, and generalization across tasks.

An essential yet often overlooked aspect of world models is their operation across multiple temporal and spatial scales. Human mental models seamlessly integrate predictions spanning milliseconds (reflexive responses), seconds (immediate action planning), minutes to hours (task completion), and even years (life planning) [366]. This multi-scale capability allows us to simultaneously predict immediate physical dynamics while maintaining coherent long-term narratives and goals. Similarly, humans process spatial information across scales—from fine-grained object manipulation to navigation across environments to abstract geographical reasoning. Current AI world models typically excel within narrow temporal and spatial bands, whereas human cognition demonstrates remarkable flexibility in scaling predictions up and down as context demands. This suggests that truly general-purpose AI world models may require explicit mechanisms for integrating predictions across multiple time horizons and spatial resolutions, dynamically adjusting the granularity of simulation based on task requirements.

One central challenge in designing world models is the interplay between **complexity** and **predictive accuracy**. As discussed, implicit models, such as those based on recurrent neural networks or transformers, offer simplicity and elegance, but they often come with the trade-off of limited interpretability. The model’s internal state is an opaque latent space, making it difficult to enforce domain constraints or provide guarantees about the accuracy of predictions. While such systems excel at capturing highly complex relationships and data-driven patterns, they also risk overfitting or failing to generalize to unseen scenarios.

Explicit models, by contrast, offer greater transparency and control. By factorizing state transitions and observations into separate functions, we gain a clearer understanding of how predictions are formed, and we can more easily integrate structured knowledge, such as physical laws or domain-specific rules. However, this approach comes with its own set of challenges. First, it often requires large amounts of labeled training data or simulated experiences to accurately capture environment dynamics. Second, even the most well-structured explicit models may struggle with complex environments that require fine-grained, high-dimensional state representations, such as in video prediction or robotics.

The **simulator-based** approach offers a promising alternative, wherein agents rely on external environments—either physically grounded or simulated—for dynamic updates. This method avoids many of the challenges inherent in learning accurate world models from scratch, as the simulator itself serves as the “oracle” of state transitions and observations. However, the reliance on simulators also introduces limitations: simulators often fail to capture the full richness of real-world dynamics and can be computationally expensive to maintain or scale. Furthermore, real-world environments introduce noise and variability that a purely learned or pre-configured model might miss. As AI agents strive to perform tasks in open-ended, unpredictable settings, the robustness of their world models will be tested by the gap between simulated and actual environments.

A key theme that emerges from this discussion is the **trade-off between generalization and specialization**. The more specific a world model is to a particular domain or task, the less likely it is to generalize across different contexts. Models like MuZero [349] and Dreamer [350] exemplify this: they excel at specific environments (e.g., Atari games or robotics) but require careful adaptation when transferred to new, uncharted domains. Conversely, implicit models—particularly those leveraging large-scale neural networks—have the potential to generalize across tasks but often do so at the cost of sacrificing domain-specific expertise.

Moreover, **integrating memory** with world models is crucial for agents that need to handle long-term dependencies and past experiences. While world models excel at predicting the next state based on immediate inputs, true intelligent behavior often requires reasoning about distant outcomes. Long-term memory allows agents to store critical environmental knowledge, ensuring that short-term predictions are grounded in a broader understanding of the world. This fusion of memory, perception, and action, mediated by the world model, creates a feedback loop where predictions shape actions, which in turn inform future predictions.

The **human analogy** remains compelling: just as humans integrate sensory inputs, memories, and internal models to navigate the world, so too must intelligent agents combine perception, memory, and action through their world models. As the field advances, it is clear that a holistic approach—one that unifies implicit, explicit, and simulator-based methods—may be the key to achieving more robust, generalizable, and adaptive agents. Hybrid methods, like those used in AutoManual [108] or discovery-based models [356], offer exciting possibilities for blending learned knowledge with structured rules and real-time interactions, potentially pushing the boundaries of what we consider a world model.

Looking forward, **open questions remain**. How can we ensure that world models exhibit **long-term stability** and **reliability** in real-world settings? How do we handle the inherent **uncertainty** in dynamic environments while maintaining the flexibility to adapt? Furthermore, as agents grow more sophisticated, how can we design systems that are both **efficient** and **scalable** across increasingly complex tasks without incurring massive computational costs?

In conclusion, the future of world models lies in their ability to balance the need for **generalization** with the requirement for **domain expertise**. By continuing to explore and refine the interplay between model simplicity and complexity, between external and internal approaches, we move closer to developing AI systems that not only understand the world but can actively shape their understanding to navigate and adapt in a rapidly changing reality.